

Wildland Fire Location Management System (WFLMS)

Robert Goldie and Jack Gallegos

Senior Project

ELECTRICAL ENGINEERING DEPARTMENT

California Polytechnic State University

San Luis Obispo, CA

June 12, 2020

Table of Contents

List of Figures	3
Acknowledgements	4
Abstract	5
Chapter 1: Introduction and Background	6
Chapter 2: Project Requirements	9
Chapter 3: Project Goals	10
Chapter 4: Hardware Design	11
Network Overview	11
Hardware Description	13
Chapter 5: Software Design	14
Chapter 6: Development and Construction	15
Hardware Development	15
Software Development	17
Chapter 7: Integration and Testing	18
Full System Test	19
Chapter 8: Conclusion	22
Results of Project	22
Future Work	22
Chapter 9: Bibliography	24
Chapter 10: Appendices	26
A: Senior Project Analysis	26
B: Project Bill of Materials	30
C: GitHub Code	30

List of Tables

Table I: WFLMS Packet Structure.....	12
Table II: Project Bill of Materials.....	30

List of Figures

Figure 1: Automatic Packet Reporting System (APRS) Transmitter For Tracking...	6
Figure 2: West Marine Automatic Identification System (AIS) Mapping Device.....	6
Figure 3: Hungarian Firefighters Avoiding Water Drop.....	7
Figure 4: Network Overview.....	11
Figure 5: GPS Tracker (Left) and Base Station (Right) Block Diagrams.....	13
Figure 6: Combined Software Flowchart.....	14
Figure 7: WFLMS Hardware (All Nodes).....	16
Figure 8: Screenshot of Mapping Interface.....	17
Figure 9: Raspberry Pi Running LoRa Example Code.....	18
Figure 10: Raspberry Pi Running GPS Monitor (gpsmon).....	19
Figure 11: Airborne Repeater.....	19
Figure 12: Airborne Repeater Close-up.....	20
Figure 13: LoRa Antenna and GPS Antenna of GPS Tracker Node on Car Roof...	20
Figure 14: System Range Testing.....	21

Acknowledgements

We would like to thank the following people for their contributions to and assistance with our project: Dr. Dennis Derickson, Professors Chuck Bland and Robert Aversano, and the various members of the Cal Poly Amateur Radio Club. Dr. Derickson has been our advisor for this project. Professors Bland and Aversano assisted with practical knowledge and advice. Thank you to everyone else who helped us along the way.

Abstract

Wildland firefighters work in complex, dangerous environments where effective communication is key to developing situational awareness. Currently, the only communication firefighters have is with Very High Frequency (VHF) 160 MHz band analog radios. While digital radio protocols could potentially allow location data and commands to be sent to field crews, the technology is primarily for voice communication. This project presents a new system that frees up voice channels and uses LoRa transceivers for robust communication. The system allows for emergency and location reports to be sent to a dispatcher via an airborne repeater. The entire system is software-defined, so it can be updated as firefighter needs evolve.

Chapter 1: Introduction and Background

Location tracking beacons are useful for tracking assets and providing situational awareness. There are many applications of this, from airplane tracking for air traffic control, to amateur packet radio stations tracking vehicles at public service events, or even maritime traffic control [1]. Figures 1 and 2 show examples of location tracking systems in use today [2, 3].



Figure 1: Automatic Packet Reporting System (APRS) Transmitter For Tracking



Figure 2: West Marine Automatic Identification System (AIS) Mapping Device

One such use case for these beacons is in wildland firefighting, where emergency responders typically work in very remote areas. With few roads or landmarks in the vicinity, it can be difficult to place crews in the correct areas in order to construct a successful fireline (a fire break in which fuels are removed from a fire's path). By giving each crew a beacon, incident command (IC) staff can continuously monitor crew location and progress in order to direct crews from the incident command post. This reduces the need for updates via voice radio communications and allows the IC to view the crew locations on a live map. This also allows firefighting aircraft to make sure crews will not be injured by falling retardant or water, preventing fatal incidents [4]. Figure 3 below shows firefighters dangerously close to a water drop. This kind of situation could be prevented if firefighters had a way of signalling position to drop operators.



Figure 3: Hungarian Firefighters Avoiding Water Drop

Sending data over separate radio networks from voice frequencies (VHF 160 MHz band) increases radio channels' capacity, leaving room for critical voice traffic. This offers improvement over digital radios that can optionally send location reports, which often require costly software/hardware upgrades to support Global Positioning System (GPS) tracking.

This also allows fire departments to keep their existing radios. A quote we found for Motorola Project 25 (P25) radios certified for use in fireground environments showed

costs of several thousands of dollars just for the radios [5]. Although the US Forest Service would like to switch to P25 in the future, there are plenty of analog radios that still have use and will be more useful once a tactical networking solution like WFLMS is integrated into emergency response operations.

In these remote areas, cellular coverage is often spotty or non-existent. Tracking can be achieved by using high-performance communications protocols such as LoRa and an airborne repeater attached to an IC aircraft. Using mesh networking could allow trackers to beat line-of-sight radio contact and allows for dynamic networking. This project seeks to use LoRa and optionally mesh networking to create an open-source and dynamic tactical communications network for wildland firefighters that can scale with incident size and complexity.

Chapter 2: Project Requirements

The WFLMS must be robust and usable for firefighters in high-stress environments. While location information is most important for this system, there are other useful pieces of information that firefighters can utilize. Emergency status information and resource status should be sent to improve the IC's situational awareness. Additionally, fireline construction (constructing or not constructing) is useful information for the IC. The time the location message was received is also useful.

The hardware devices should be easy to use and require minimal training/setup. We use the buttons on the LoRa hat boards to simplify the user input process, however in a full-fledged version these could be weather-sealed buttons accessible outside of an enclosure. The web page developed for the IC to view unit locations is very simple and can run on computers with modest computing power. It has an intuitive user interface and auto-refreshes unit locations as they come in, allowing the IC to focus on other communication tasks. Another design requirement is that the system can be portable not only for vehicles, but also for handcrews, who routinely hike many miles per day and even camp in remote areas. Finally, the system needs to be low-cost to enable rapid prototyping which would allow us to show proof-of-concept.

Chapter 3: Project Goals

The project has two primary goals (1, 2) and two stretch goals (3, 4):

1. *Display crew locations on a monitor for IC and aircraft operators' situational awareness.*
2. *Use a repeater to increase communications range beyond line of sight.*
3. *Stretch goal: Display coordinates as features on a map.*
4. *Stretch goal: Use mesh networking to efficiently route communication.*

The first project goal requires each GPS-enabled Raspberry Pi to send GPS coordinates over LoRa to the command post Raspberry Pi. This requires a GPS module and antenna, as well as a remote power source. At the receiver end, the GPS coordinates will be printed out in the terminal without a map. This goal is the minimum viable product (MVP) for the project.

The second project goal uses a Raspberry Pi to extend the range of the system by placing it above the incident scene to operate as a repeater. This allows for better line-of-sight for the various ground units, while also extending the range of the LoRa communications from the Raspberry Pi in the air to the incident command post Raspberry Pi.

The third project goal is to display the crew GPS coordinates as features on a map, rather than as text. This will allow the operator to see a real-time view of the crew locations, rather than needing to input the GPS coordinates into a separate system to interpret them.

The fourth project goal is very much a stretch goal, implementing mesh networking. This would allow for unit-to-unit communications without routing through the repeater, if possible. This is a more efficient way of communicating while also reducing the burden on the repeater. However for this to be implemented there will need to be information associated with the routing path and an algorithm will need to be developed.

Chapter 4: Hardware Design

Network Overview

The WFLMS network consists of ground transmitter nodes that can send data to IC staff or to other ground units through an airborne repeater. The airborne repeater re-transmits the data so that the IC and others can receive it beyond line of sight. Figure 4 shows how the system exploits the aerial command aircraft's position (almost always present on fires where knowing crew locations would be useful) to repeat data to other units assigned to the incident.

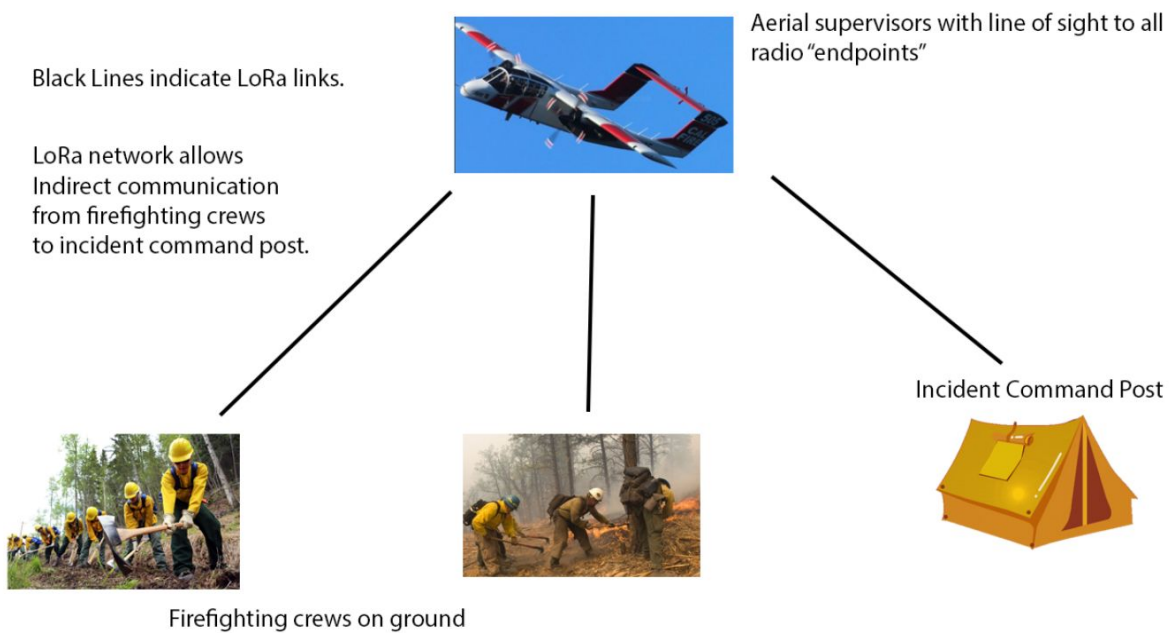


Figure 4: Network Overview

In order to send location data through the WFLMS network it was necessary to create a custom packet format to store the data. Although we use LoRa radios [6] for interoperability in this particular implementation, the platform has not been defined at the physical layer so other types of radios/modulation schemes could be used. The WFLMS packet handles the Application/Presentation layer of the Open Systems Interconnection (OSI) Model [7] and is 20 octets long.

TABLE 1: WFLMS Packet Structure

Bit	Field	Data Type in Packet object	Encoding in Lora Packet
156-159 (LSB)	N/A	N/A	Unused bits
152-155	SRC_TYPE	4-Bit Integer	Source Unit Type. 0 - Handcrew 1 - Engine Crew 2 - Strike Team 3 - Dozer 4 - Water Tender 5 - Medical Team 6 - Single Engine Air Tanker 7- VLAT 8 - Helicopter 9 - Air Attack Supervisor (ATGS) 10 - Incident Command (IC) 11 - 15- Reserved for future use
144-151	SRC_NUM	8-Bit Integer	Source Unit Number
140-143	DEST_TYPE	4-Bit Integer	Destination Unit Type Same as Source Type
132-139	DEST_NUM	8-Bit Integer	Destination Unit Number
4-131	MGRS_LOC	16 bytes, encoded as ASCII String	GPS Location encoded using the Military Grid Reference System (MGRS). Only the lowest 15 bytes (chars) are used, per "mgrs" library.
2-3	RSRC_STAT	Bits 0-1 of Integer	Resource Status 0 = Active 1 = On break 2 = In transit 3 = Out of service
1	FLINE_STAT	integer	Fireline Construction Status (For hand crews only) 0 = No 1 = Yes
0 (MSB)	EMERG_FLG	Bit 0 of Integer	Emergency flag 0 = No Emergency 1 = Emergency

The 160-bit packet defined in Table 1 allows for location information as well as vital crew information to be transmitted as a single packet. The LoRa board can send up to 255 bytes per transmission [8], so this packet could be expanded to fit future needs.

Hardware Description

The 900-928 MHz Industrial, Scientific, and Medical (ISM) band was chosen because these frequencies do not require a license to use them. This means that transmissions may receive interference from other devices in this band, but the robustness of the LoRa protocol allows the system to perform even in very low signal to noise ratio (SNR) scenarios [9].

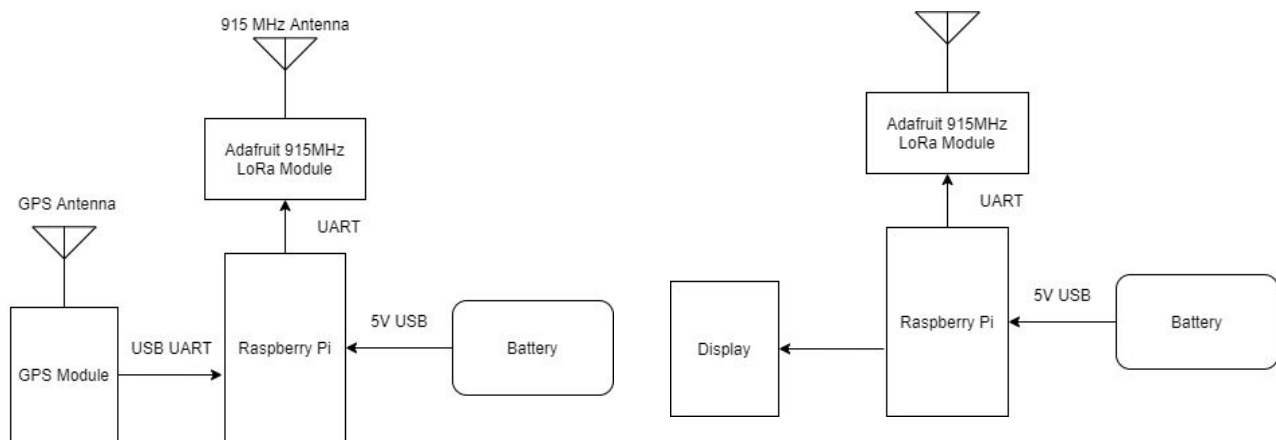


Figure 5: GPS Tracker (Left) and Base Station (Right) Block Diagrams

The ground units have GPS-enabled Raspberry Pi's which transmit their coordinates to the repeater or incident command post (see software flowchart) using an attached radio module. The units operate with portable 5V USB battery packs. Figure 5 shows the GPS Tracker and Base Station node block diagrams. The command post node consists of a Raspberry Pi connected to a monitor which shows the map with real time locations of the ground units. It also can run off a 5V battery.

The airborne repeater node is the same as the GPS tracker node, but without a GPS module (just a Raspberry Pi and LoRa hat). It can fit on anything from a small quadcopter to helicopters or full size command aircraft. The Raspberry Pi uses more power than a microcontroller driven device (like an Arduino), so a high-capacity battery is needed to ensure the repeater runs constantly.

Chapter 5: Software Design

This software flowchart in Figure 6 depicts a high-level overview of the code running on each Raspberry Pi. With many ground units transmitting on the same frequency, it is important to have the ability to throw out bad packets, something that the LoRa protocol handles for us. Additionally, the system is currently configured to pass information from ground unit, to repeater, to base station, however, in the future it may be desired to have information flow between crews as well.

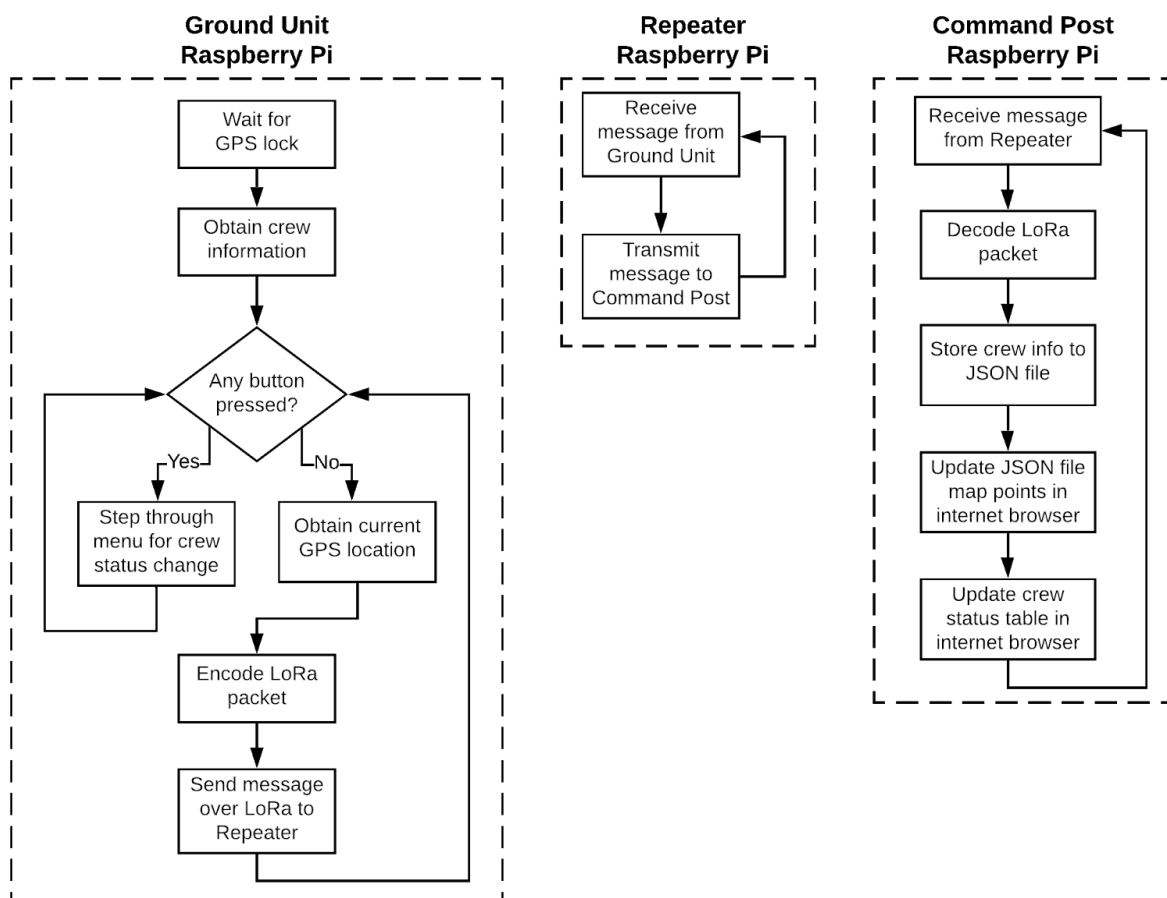


Figure 6: Combined Software Flowchart

Chapter 6: Development and Construction

Hardware Development

We decided to use the Raspberry Pi single board computer (SBC) platform for all nodes because it allows us to use USB peripherals and the Pi's built in Input/Output (I/O) pins. Raspberry Pis are very popular in the electronics community, allowing people to collaborate as well as learn from projects posted on public websites. With many GPS projects publicly available, Raspberry Pis have been proven capable of handling the complexity of this project. Additionally, while the Arduino and Texas Instruments MSP microcontrollers were used in several classes at Cal Poly, there were less examples of GPS projects using it, leading to concern over whether the project would be too complex to achieve on a microcontroller. For our particular project we used two Raspberry Pi 3 Model B SBCs for the GPS tracker and repeater nodes, and a Raspberry Pi 4 Model B (later switching to a Pi Zero W) for the incident command post station.

The next component chosen was the Adafruit RFM95 LoRa RF Hat (Adafruit product ID 4074) to facilitate the sending and receiving of data. Adafruit's published working libraries make integration much simpler and maintain the focus of the project on integrating off the shelf hardware rather than writing custom drivers. The libraries also handle things like Forward Error Correction (FEC) at a lower level so that we could focus on sending and receiving data. Supporting components like the 900 MHz antennas, cables, and battery packs were chosen primarily based on function and cost. Finally, late additions such as active GPS antennas allowed us to improve the likelihood of and speed of getting GPS lock. The addition of the Pi Zero for a lightweight repeater node helped reduce power consumption and extended drone flight time.

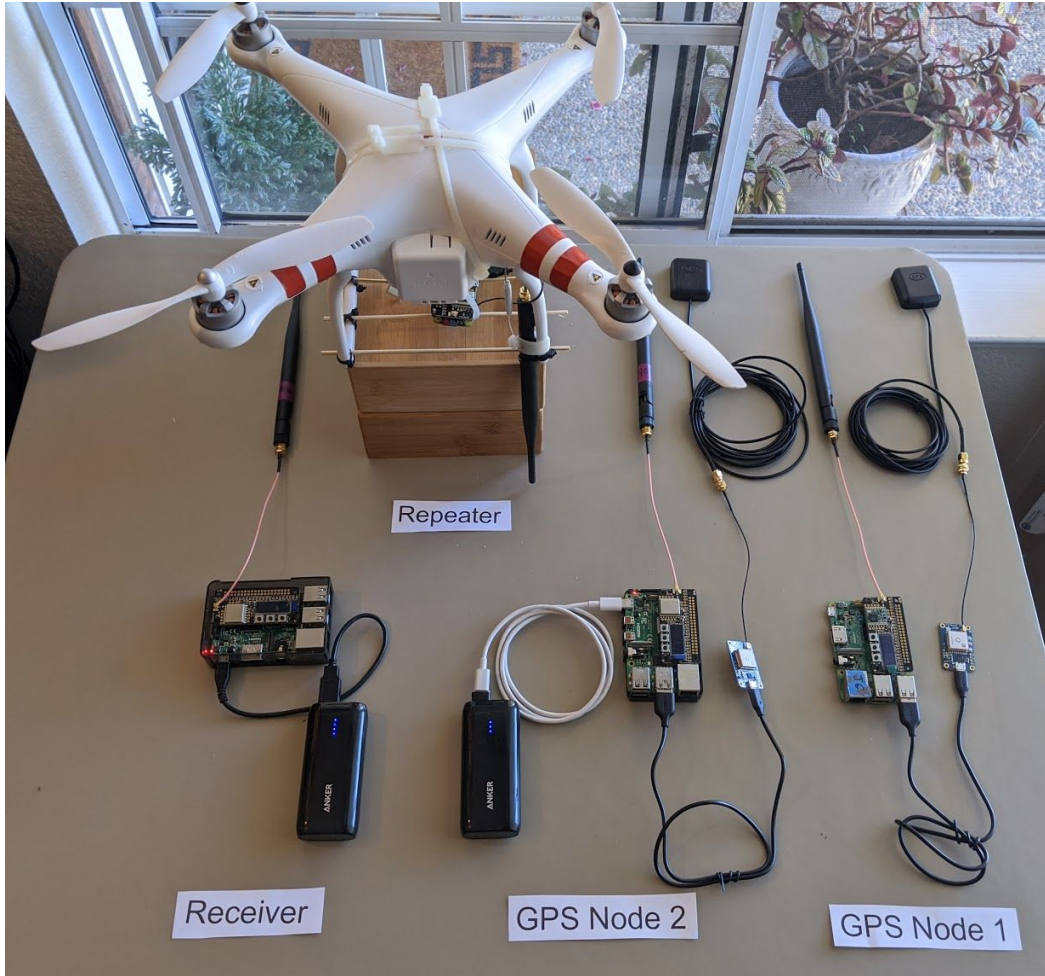


Figure 7: WFLMS Hardware (All Nodes)

Project construction was fairly simple, with all components being chosen for their compatibility. This included connecting each Raspberry Pi to a battery, a LoRa board, an antenna, and a GPS antenna (for the location-tracking Pis). Additionally, a drone was used to elevate the Repeater Pi for better line of sight. This included several zip ties in order to secure the payload to the drone. While there were no enclosures for the Raspberry Pis, a future design modification would be to create compact enclosures to eliminate excessive wires for each Pi. This would also allow for better testing as the setups would be less fragile. Figure 7 shows the assembled hardware with each type of node displayed.

Software Development

The Raspberry Pi platforms use Adafruit's CircuitPython libraries [8], including those for the RFM95 LoRa hat. The overall programming was very simple and just required gathering the data to be sent (through the push buttons on the LoRa hats) and the location (through the GPS). The transmitter software routines then assemble all the relevant status information into the packet format and send it. The repeater simply re-transmits any packets received that are addressed to the repeater.

The mapping interface takes in the relevant unit information JSON file and uses the Google Maps Application Programming Interface (API) to handle mapping. The node.js application "http-server" hosts the web server. The mapping interface displays the unit's label and location, as well as other information that could help IC staff. Figure 8 shows the mapping interface including pins for active crews as well as their statuses in a table at the bottom of the webpage.

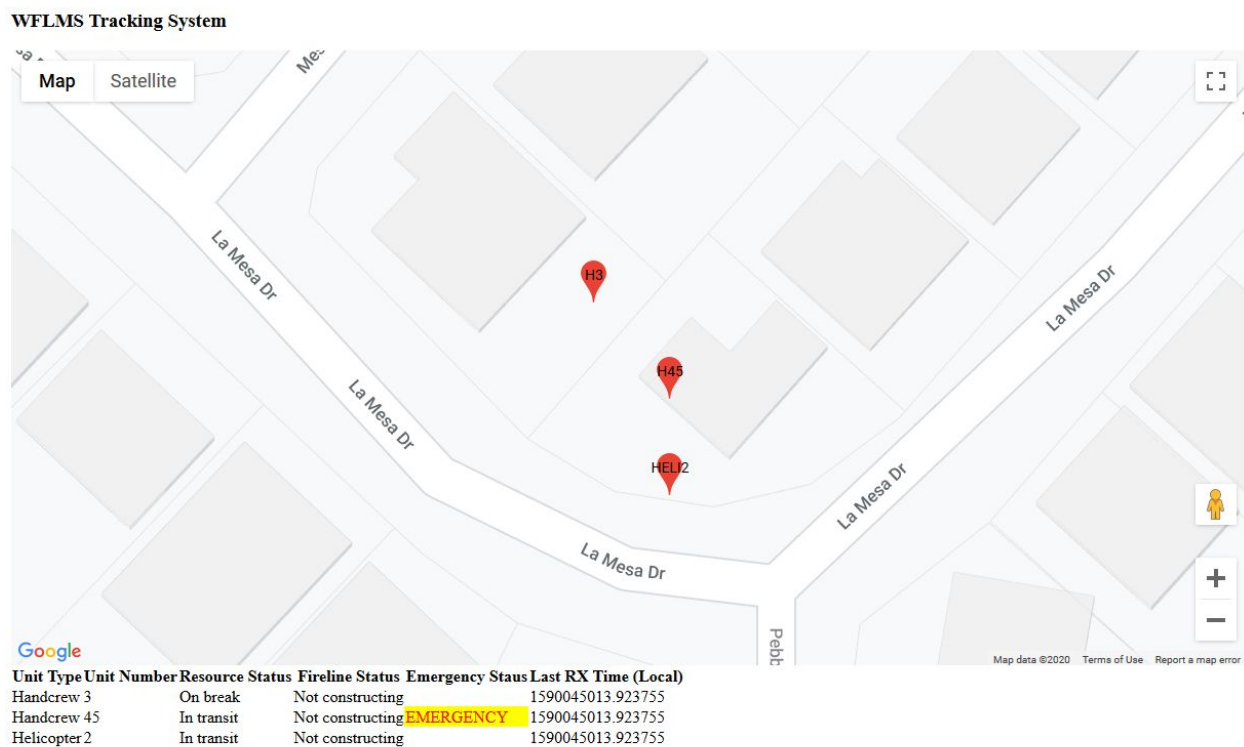


Figure 8: Screenshot of Mapping Interface

Chapter 7: Integration and Testing

Integration

Integration occurred in several stages as new components were brought online. This allowed for easier debugging as pieces were tested independently and problems could be better isolated.

The first stage of testing was devoted to testing the LoRa radios with the distributor provided libraries. We first ran example code to verify correct operation of the radios and then began to modify the code and familiarize ourselves with the library. Figure 9 shows the Raspberry Pi running the example code. The onboard LED screen was useful for debugging and was used to display when the radio was ready to receive a packet.

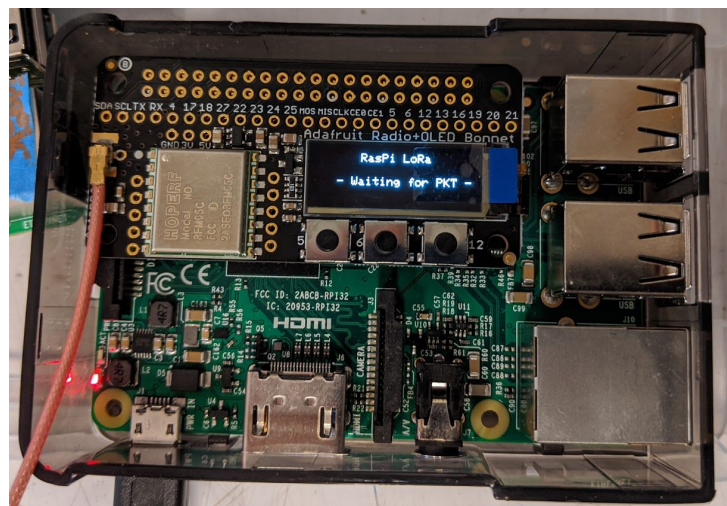


Figure 9: Raspberry Pi Running LoRa Example Code

The next stage focused on the GPS module and GPS coordinate formats. We again downloaded the necessary packages to the Pis and then ran a generic GPS monitor (gpsmon) to familiarize ourselves with the data it provided. Figure 10 shows the monitor running in the terminal on a Pi.

Time: 2020-06-07T03:10:45.000Z Lat: 37 28' 51.06599" Non: 122 16' 44.03940" W										
Cooked TPV										
GPGGA GPGSA GPRMC GPVTG GPGSV										
Sentences										
Ch	PRN	Az	El	S/N	Time: 031045.000			Time: 031045.000		
0	20	43	57	35	Latitude: 3728.8610 N			Latitude: 3728.8610		
1	27	286	52	37	Longitude: 12216.7399 W			Longitude: 12216.7399		
2	32	189	40	33	Speed: 0.16			Altitude: 175.7		
3	18	0	0	33	Course: 127.79			Quality: 1 Sats: 03		
4	21	0	0	34	Status: A FAA: A			HDOP: 1.92		
5					MagVar:			Geoid: -25.7		
6					RMC			GGA		
7										
8					Mode: A2 Sats: 20 27 32			UTC: RMS:		
9					DOP: H=1.92 V=1.00 P=2.16			MAJ: MIN:		
10					TOFF: 0.124364991			ORI: LAT:		
11					PPS:			LON: ALT:		
GSV					GSA + PPS			GST		

Figure 10: Raspberry Pi Running GPS Monitor (gpsmon)

We then parsed the data in order to receive only the GPS coordinates periodically. The next step was converting the degrees minutes seconds GPS format into the Military Grid Reference System (MGRS). We then moved to full system tests to validate our programs.

Full System Test

The second stage of testing was to do a full system test. The airborne repeater consists of a DJI Phantom 2 quadcopter attached to a repeater node. The airborne repeater hovered around 100 feet off the ground. Figure 11 shows the airborne repeater and Figure 12 shows a closeup of the components onboard.



Figure 11: Airborne Repeater

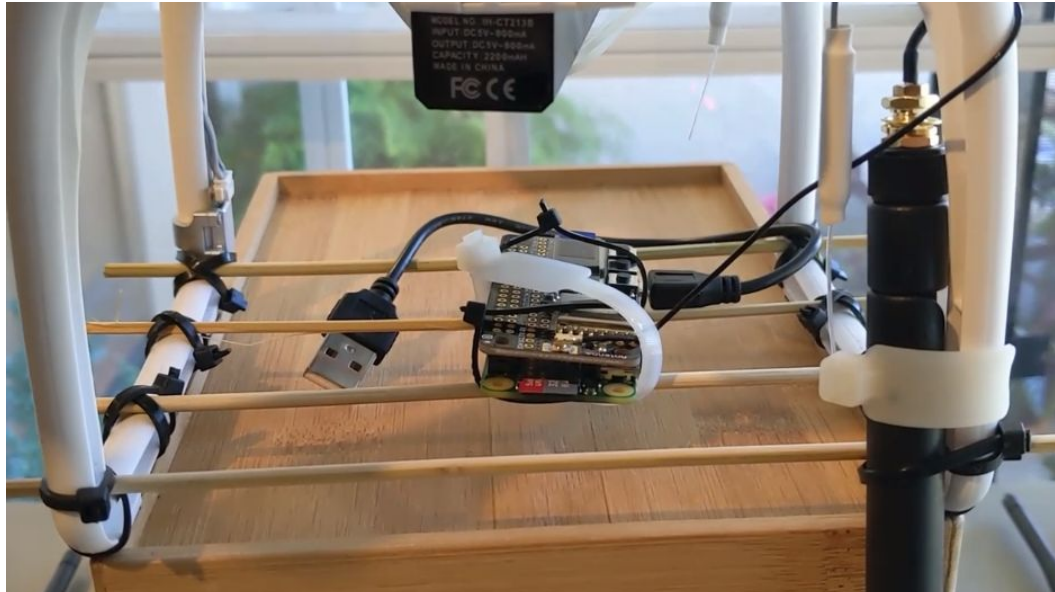


Figure 12: Airborne Repeater Close-up

We then took one of the GPS nodes and put it on a car. The car drove around the test area. Figure 13 shows the external antennas mounted to the car roof.



Figure 13: LoRa Antenna and GPS Antenna of GPS Tracker Node on Car Roof

We successfully received packets while the car drove around. Figure 14 shows the location mapping software running while the system is active.

WFLMS Tracking System

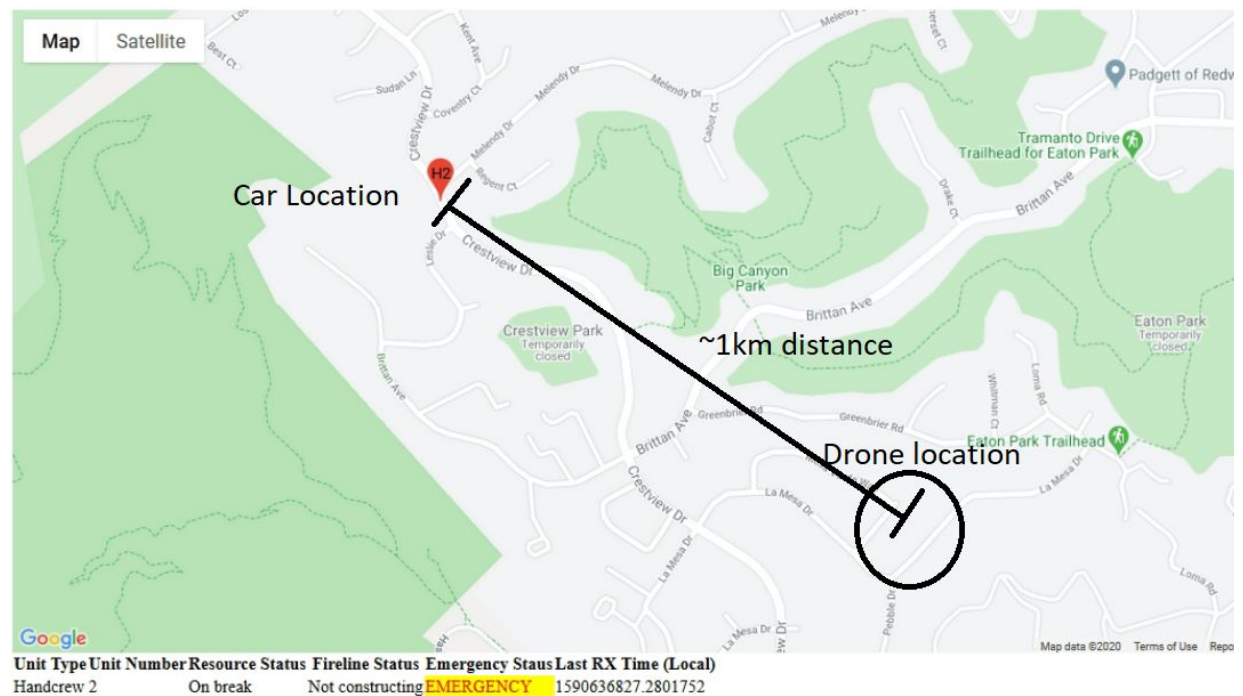


Figure 14: System Range Testing

Figure 14 also shows the furthest communication we had within the system, approximately 1 km. This was due to the density of houses in the neighborhood in which we tested the system. The actual use case is typically in open wildland which would likely increase the range considerably. Due to shelter-in-place orders, we were not able to test the system in an environment that would typically be found on a wildfire.

Chapter 8: Conclusion

Results of Project

The system did work as intended and we were able to meet one of our stretch goals (display coordinates as features on a map). Our successful test can be found in the demo video on YouTube at: <https://youtu.be/hHjlfwgMbhl> The drone enabled us to reach a system range of 1 km because without it, there would not have been line-of-sight between the radios operating on simplex. Additionally, we were able to show the crew statuses on the webpage along with the map features. However, the system did have stability issues as it required multiple attempts in order to have a successful run. The client side web application we developed was caching the JSON file and ended up having to be forced to not cache in order to work. Additionally, there were occasional packet errors that resulted in parts of the system crashing when decoding the packet fields. We also had issues with receiving packets reliably. This can be attributed largely to the non-ideal environment in which we tested the system (reduction in received signal strength due to obstructions). On a wildland fire, there are typically very little to no structures in the area. The structures in the test environment limited the repeatability of tests and required us to fly the drone as high as possible in order to maximize system coverage. Additionally, without enclosures, the Raspberry Pis were very delicate, especially with the u.FI connectors. This introduced another source of error into the project. In conclusion, we have built the framework for a system that could improve communications situational awareness on wildland fire incidents. The system could be improved and brought up to scale and tested in a wildfire deployment scenario.

Future Work

We showed how the system can work in a limited capacity with one or two transmitters and a basic aerial repeater. Future improvements could include:

- Defining standards for other layers of the system to make sure other nodes are interoperable on fires (perhaps using LoRa or other high performance protocol)
- Add more data fields to the protocol to allow for mesh networking and actual tactical networking
- Investigating throughput and system capacity on high profile incidents

- Creating a self contained transmitter / repeater / ground station that is ready to deploy at scale at incidents
- Create enclosures for devices to resist smoke/weather conditions
- Improve the ground station to improve logging, display, and data sharing capabilities
- Fix error handling and solve issues with position encoding/decoding and make the system stable
- Automate node startup and shutdown procedure
- Utilize power saving measures on the Raspberry Pis when not active

Chapter 9: Bibliography

[1] "Automatic identification system," 30-May-2020. [Online]. Available: https://en.wikipedia.org/wiki/Automatic_identification_system. [Accessed: 11-Jun-2020].

[2] "Automatic Packet Reporting System," Wikipedia, 26-May-2020. [Online]. Available: https://en.wikipedia.org/wiki/Automatic_Packet_Reporting_System. [Accessed: 11-Jun-2020].

[3] "Be Safe and Be Seen with AIS," West Marine. [Online]. Available: <https://www.westmarine.com/WestAdvisor/Be-Safe-and-Be-Seen-with-AIS>. [Accessed: 11-Jun-2020].

[4] CAL FIRE/USFS Aerial Firefighting Drop Safety Video. 'CAL FIRE TV' Youtube Channel, 2019.

[5] "Motorola Solutions Portable Radios and Accessories Pricing Sheet," www.charlottealliance.org. [Online]. Available: <http://www.charlottealliance.org/pdf/motorola/2017000064%20-%20Pricing%20-%20Motorola%20Solutions%20-%20Portable%20Amend%201.pdf>. [Accessed: 10-Jun-2020].

[6] Adafruit Industries, "Adafruit LoRa Radio Bonnet with OLED - RFM95W @ 915MHz," [adafruit industries blog RSS](https://adafruitindustriesblog.wordpress.com/2017/05/adafruit-lora-radio-bonnet-with-oled-rfm95w/). [Online]. Available: <https://www.adafruit.com/product/4074>. [Accessed: 11-Jun-2020].

[7] "OSI model," Wikipedia, 27-May-2020. [Online]. Available: https://en.wikipedia.org/wiki/OSI_model. [Accessed: 11-Jun-2020].

[8] "adafruit_rfm9x CircuitPython documentation," [adafruit_rfm9x - Adafruit RFM9x Library 1.0 documentation](https://circuitpython.readthedocs.io/projects/rfm9x/en/latest/api.html). [Online]. Available: <https://circuitpython.readthedocs.io/projects/rfm9x/en/latest/api.html>. [Accessed: 11-Jun-2020].

[9] "LoRa Documentation," LoRa Documentation by Eric B. [Online]. Available: <https://lora.readthedocs.io/en/latest/>. [Accessed: 11-Jun-2020].

[10] "Study: Fire seasons getting longer, more frequent – Climate Change: Vital Signs of the Planet," NASA, 03-Aug-2015. [Online]. Available: <https://climate.nasa.gov/news/2315/study-fire-seasons-getting-longer-more-frequent/>. [Accessed: 11-Jun-2020].

Chapter 10: Appendices

A: Senior Project Analysis

Project Title: Wildland Fire Location Management System (WFLMS)

Student's Name: Jack Gallegos, Bobby Goldie

Advisor's Name: Dr. Dennis Derickson Advisor's Initials: DD Date: 6/6/2020

1. Summary of Functional Requirements

The location management system tracks wildland firefighter crews in remote areas to provide incident command with real-time updates on crew locations. The system is low-cost and uses portable transceivers for better usability. A repeater is also used to increase system coverage. Crew data fields are sent along with GPS coordinates.

2. Primary Constraints

Anticipated problems include displaying GPS coordinates with a mapping software and achieving desired system range. The mapping poses a challenge because it will require interfacing new data formats and coding languages which we have not previously used. Additionally, the system range will be affected by many factors, some of which can be controlled in our design and testing of the system. A final constraint will be working on the project with virtual resources due to the COVID-19 situation. This will require the members of the project to collaborate in new ways and may slow down progress and testing.

3. Economic

The economic impacts of the system are primarily firefighting agencies' budget for new technology. These agencies do not typically have excess funds due to the growing length and intensity of the fire season [10]. However, new technology such as crew location tracking has the potential to save lives. While the loss of life cannot be compared to a dollar amount, a low-cost system such as this one could provide agencies with an extra layer of security to enhance the safety of firefighters.

At present, the system is using a publicly available frequency (915 MHz) for LoRa communications. Public agency use potentially has no costs for licensing its own

frequency since FCC does not charge for public agency use of the spectrum. The estimated component cost is \$200 per unit. While the project was planned to take three quarters there were many obstacles resulting in only one quarter for design and testing of the system in its current form.

The primary users of the system are firefighter agencies however the system could be expanded for use by other public service agencies that require location tracking of units in a temporary environment.

4. If manufactured on a commercial basis

The large number of agencies not using this type of system represents a large potential market for the device. There are relatively few fire agencies that work on wildfires, typically one for each state as well as federal agencies that protect federal land. However, there is still a large market due to the growing number of firefighters each year. At an estimated 100 devices per year and \$5000 manufacturing/labor cost per device, a \$7500 purchase price gives \$2500 in profit per device. This is an estimated \$2.5 million in profit per year. The user cost is hard to estimate, but the cost of electricity to operate the device should be negligible in comparison to the purchase cost.

5. Environmental

The raw materials required to make the electronics and manufactured components in the system all have an ecological impact in the chain of production and disposal, but this is very hard to quantify because the sources of the components and materials used in subsystem manufacture are unknown.

The system has the potential to save fuel and emissions from helicopters due to a reduction in the number of emergency airlifts. This could be a result of incident command having a better idea of where crews are located and being better able to warn them of changes headed their way. Less emergencies allows for more efficient use of limited resources on a fire.

6. Manufacturability

Assembly is fairly simple as most components are store-bought, complete subsystems that are integrated together. Software needs to be flashed on the Raspberry Pi's but this process can be easily automated. The only challenge for assembly is assembling everything into a compact, water-tight enclosure. This would likely require hands-on assembly, increasing the cost of production. Components need

to be sourced from reliable vendors and quality control tests need to be developed to validate the correct assembly and operation of each unit.

The supply chain can be simplified by sourcing large subsystems items (such as Raspberry Pi's and LoRa radios) from their respective manufacturers rather than from retail distributors. This saves cost and simplifies logistics with ordering large quantities.

Distribution is handled online to reduce overhead required to sell the product. One challenge is marketing to fire agencies. This can be accomplished by contacting fire agencies and offering free trials. Additionally, storing finished products will require a warehouse or storage unit, adding cost and complexity to the distribution process.

7. Sustainability

Using Raspberry Pis allows the system to be expanded when more features are desired or new hardware is made available. It also allows the system to grow to the size needed so there is little wasted equipment. This contributes to a sustainable system in which there are few wasted resources and taxpayer money is used most efficiently. Although the system is relatively future proof, the location tracking may need to be updated as new protocols are released and the system can be improved.

8. Ethical

The system, while intended to increase safety on the fireline, has the potential to fail. This would result in the incident commander relying on less accurate forms of location tracking, such as over voice radio frequencies. According to the "Golden Rule" principle, you should treat others as you wish to be treated. Applying this to WFLMS means the system should be as robust as possible in order to reduce the chance of a system failure. Most people would want the safest system possible so the system should be designed with this in mind.

This project serves to support the first IEEE decree for ethics: "To hold paramount the safety, health, and welfare of the public" [18]. The main goal of the project is to provide firefighters with another resource to help them complete a dangerous job. This follows closely with the IEEE code of ethics which states that public safety is a top priority. One potential ethical issue with the project is that it might be insecure with respect to crew locations being broadcast over the incident area. However, wildland fires are usually in remote areas or if in urban areas, it is usually evacuated. While this is a potential vulnerability of the system, it is not one that would likely be exploited.

9. Health and Safety

Firefighting is inherently dangerous. However, this system has the potential to help mitigate some of the risks. There is still a chance that the system will fail at a critical time, potentially endangering the lives of firefighters. This is the main health and safety risk of the system: overdependence on the location management system. To reduce this risk, the system will be tested during the manufacturing process. Additionally, other means of communication should be used as a back-up in the field.

10. Social and Political

This project primarily impacts firefighting public agencies. While there are no immediate drawbacks to the system, asking for a larger budget to fund prototype projects would require political support. As the only customers are government agencies, the technology would need to be very convincing in order to devote financial resources to it and away from other firefighting resources like air support. While the project may be successful, it will ultimately require a mix of promising results and political support to be adopted into the firefighting resource pool.

Additional stakeholders include Cal Poly and the Electrical Engineering Department. Demoing the project to government representatives and researchers can increase Cal Poly's reputation in the RF research world and potentially form new partnerships. Future work at Cal Poly could expand on the project's capabilities.

11. Development

This project requires learning about firefighting practices, RF communications, and several programming languages. We also need to learn about the limitations of the LoRa protocol and the GPS protocol. Finally, testing will need to be done to determine the final specifications of the system.

B: Project Bill of Materials

TABLE 2: Project Bill of Materials

Item	Sub-system	Unit Cost	Quantity	Extended Cost
Raspberry Pi 3	All	\$30	3	\$90
LoRa Board	All	\$32	3	\$94
u.FI Pigtail	All	\$2	4	\$8
GPS Active Antenna	GPS Node	\$14	1	\$14
Anker 5V Battery	All	\$12	3	\$36
915 MHz Antenna	All	\$8	3	\$24
SMA extension cables	All	\$8	3	\$24
SD card	All	\$8	3	\$24
Misc. USB Cables	All	\$2	\$4	\$8
Total:				\$324

As Table 2 shows, the project cost \$324 for three nodes (the minimum required to demonstrate the system). With a unit cost of ~\$100, the system could easily be scaled up for larger-scale testing in order to secure sales to firefighting agencies.

C: GitHub Code

All the code for the project can be found at www.github.com/jeak/senior_project.